ESD-TR-81-302

12

Semiannual Technical Summary

Packet Speech Systems Technology

30 September 1981

Prepared for the Defense Advanced Research Projects Agency
under Electronic Systems Division Contract F19628-80-C-0002 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS

Approved for public release; distribution unlimited.

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. This work was sponsored by the Defense Advanced Research Projects Agency under Air Force Contract F19628-80-C-0002 (ARPA Order 3673).

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Raymond L. Loiselle

Raymond L. Loiselle, Lt.Col., USAF
Chief, ESD Lincoln Laboratory Project Office

(12)

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LINCOLN LABORATORY

PACKET SPEECH SYSTEMS TECHNOLOGY

SEMIANNUAL TECHNICAL SUMMARY REPORT TO THE DEFENSE ADVANCED RESEARCH PROJECTS AGENCY

1 APRIL — 30 SEPTEMBER 1981

ISSUED 4 JANUARY 1982

Approved for public release; distribution unlimited.

LEXINGTON MASSACHUSETTS

ABSTRACT

This report describes work performed on the Packet Speech Systems Technology Program sponsored by the Information Processing Techniques Office of the Defense Advanced Research Projects Agency during the period 1 April through 30 September 1981.

CONTENTS

INTRODUCTION AND SUMMARY

The long-range objectives of the Packet Speech Systems Technology Program are to develop and demonstrate techniques for efficient digital speech communication on networks suitable for both voice and data, and to investigate and develop techniques for integrated voice and data communication in packetized networks, including wideband common-user satellite links. Specific areas of concern are: the concentration of statistically fluctuating volumes of voice traffic, the adaptation of communication strategies to varying conditions of network links and traffic volume, and the interconnection of wideband satellite networks to terrestrial systems.

Previous efforts in this area have led to new vocoder structures for improved narrowband voice performance and multiple-rate transmission, and to demonstrations of conversational speech and conferencing on the ARPANET and the Atlantic Packet Satellite Network.

The current program has two major thrusts: i.e., the development and refinement of practical low-cost, robust, narrowband, and variable-rate speech algorithms and voice terminal structures; and the establishment of an experimental wideband satellite network to serve as a unique facility for the realistic investigation of voice/data networking strategies.

This report covers work in the following areas: digital LSI vocoder development; embedded CVSD-based (ECVSD) speech waveform encoder implementation; development and experimental tests of modular packet voice terminals (PVTs) and local access area (LEXNET) facilities; development of a miniconcentrator facility to serve as a gateway between the LEXNET and the wideband satellite network (WB SATNET), and execution of packet speech experiments using this facility; and definition and planning of, and participation in, experiments on the wideband integrated voice/data network.

A manufacturer-supplied real-time emulator/debugger has been used to successfully test the Gold pitch detector for the single-card channel vocoder based on a soon-to-be-available commercial LSI signal-processing interface (SPI) chip. A single-card LPC vocoder based on the same chip has been designed, a single unit has been fabricated, and SPI microcode is under development. Two ECVSD cards have been constructed and debugged, and the required microcode has been developed and tested in conjunction with a PVT.

Ten PVT units have been constructed and debugged, and a standalone EPROM version of the PVT has been developed. PVT technology transfer has been initiated by setting in motion an "expression-of-interest" exercise with potential vendors, and by preparing a detailed technical information package on the PVT. PVT software has been expanded to include source-routing and the ability to communicate through non-stream gateways by encapsulating stream (ST) packets in Internet Protocol (IP) packets. Initial LEXNET/Voice Funnel integration has been established via a packet speech loop test involving a pair of PVTs and a LEXNET installed at Bolt Beranek and Newman (BBN).

The miniconcentrator is now operational as a flexible internet gateway, and experimental packet speech connections to LEXNETs, the WB SATNET, and the ARPANET have been demonstrated. PCM speech has been transmitted through the gateway in a three-network configuration involving two LEXNETs and the WB SATNET. The PDP-11 code for the gateway has been reorganized to provide a more effective structure for traffic control under heavy loading, and to allow more efficient scheduling of foreground and background tasks.

Lincoln has continued to play an active role in the coordination of activities aimed at checking out and characterizing the wideband satellite link. Some two-site tests involving the demand-assignment processors (PSATs) at Lincoln and Information Sciences Institute (ISI)

were carried out in September 1981, and these tests are continuing. We are directing current attention at locating a troublesome and unknown source of RF interference at ISI.

PACKET SPEECH SYSTEMS TECHNOLOGY

I. DIGITAL LSI VOCODERS

A. SUMMARY

Progress continues in the development of the Nippon Electric Company (N.E.C.) signal-processing interface (SPI)-based single-board channel vocoder. The microcode for the six N.E.C. μPD7720 SPI devices has been written and debugged. Several sentences of speech have been analyzed and synthesized through a non-real-time simulation of the six μPD7720 processors. A unit of the channel vocoder 7- × 7-in. wirewrap board has been fabricated and debugged. The EVAKIT-7720, a real-time emulator/debugger for the N.E.C. SPI device with external RAM replacing internal μPD7720 instruction and coefficient ROMs, has been obtained on loan from N.E.C. Microcomputers in Natick, Massachusetts. The code for the single-chip implementation of the channel vocoder's Gold pitch detector has been downloaded into the EVAKIT-7720 using the channel vocoder prototype board as the target system to obtain a real-time display of the analysis pitch track. Significantly, this not only corroborates the correctness of the SPI implementation of the pitch tracker, but also of the Lincoln Laboratory assembler, NAS, and non-real-time simulator, NECSIM on which the '7720 microcode was developed. Since delivery of the EPROM versions of the N.E.C. SPI device has been delayed, use of factory programmed ROM versions of the '7720 for a narrowband vocoder realization is being investigated. It has previously been reported that a full-duplex 8-kHz LPC implementation requires three μPD7720 devices: one for the linear predictive analysis, one for the Gold pitch detector, and one for the synthesizer. Also, it is expected that the N.E.C. '7720 ROM resources are sufficient to house both LPC analysis and synthesis programs and coefficients on a single chip, resulting in the need for a total of two mask types for a ROM implementation. Due to the smaller number of masks required, an LPC vocoder is also being developed for the ROM vocoder realization. The hardware

design for the LPC vocoder has been completed and a copy of the wirewrap board has been fabricated. A total of 21 packages are required, occupying less than half the available area of the 7- × 7-in. Augat wirewrap board. The 10-kHz channel vocoder implementation of the Gold pitch detector can be used in the LPC vocoder but must be modified to operate at the lower 8 kHz sampling rate. The microcode for the remaining two N.E.C. chips in the LPC vocoder has yet to be written, although the critical timing and memory constraints have been analyzed.

B. LSI CHANNEL VOCODER DEVELOPMENT

The μPD7720 memory usage for the six N.E.C. processors in the channel vocoder is shown in Fig. 1. The Gold pitch detector requires nearly all of the internal program memory and data RAM, while the analysis and synthesis chips use more moderate percentages of memory.

112775-N

	PERCENTAGE OF:		
	DATA RAM (128 x 16)	PROGRAM ROM (512 x 23)	DATA ROM (512 x 13)
GOLD PITCH DETECTOR	93	87	9
ANALYZER (7 of 19 channels)	52	26	21
SYNTHESIZER (10 of 19 channels)	64	38	25

Fig. 1. Channel vocoder memory requirements.

A copy of the channel vocoder wirewrap board has been fabricated and debugged (Fig. 2). A test jig for the digital vocoder board has also been designed and constructed. The jig is structured to simplify the potentially difficult problem of debugging a multi-processor system by allowing the LDSP to emulate in real time one or more of the vocoder's seven processors. For example, a possible initial debugging scenario would include one N.E.C. chip implementing the Gold pitch detector with the LDSP implementing the remainder of the channel

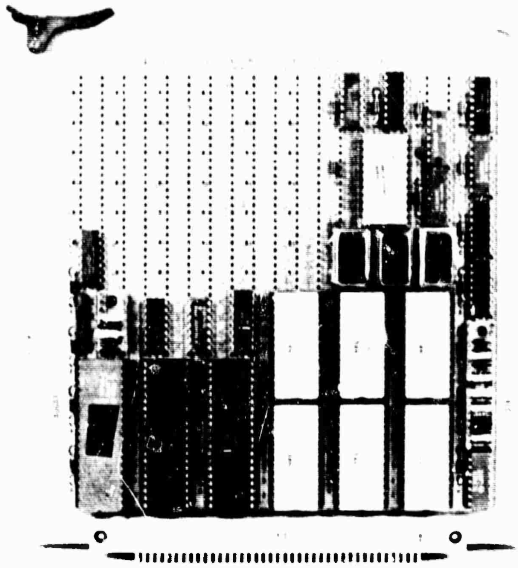

Fig. 2. N.E.C. μPD7720-based channel vocoder.

vocoder algorithm (synthesis and spectral analysis) as well as the Intel microcomputer control and communication task. This approach allows a real-time "A-B" comparison since the entire channel vocoder is already implemented in the LDSP. A next stage could be the addition of a second N.E.C. device implementing 7 of the 19 analysis channels with the LDSP processing the remaining 12 spectral channels and so on until all seven processors are debugged.

N.E.C. Microcomputers has received from Japan several units of the "EVAKIT-7720," a real-time emulator/debugger for the μPD7720 SPI. This software development tool contains a 100-pin version of the '7720 integrated-circuit die with bondouts to external RAM in place of the internal program and coefficient ROMs, as well as a microprocessor-based monitor with debugging features including user-specified program counter breakpoints, single-step

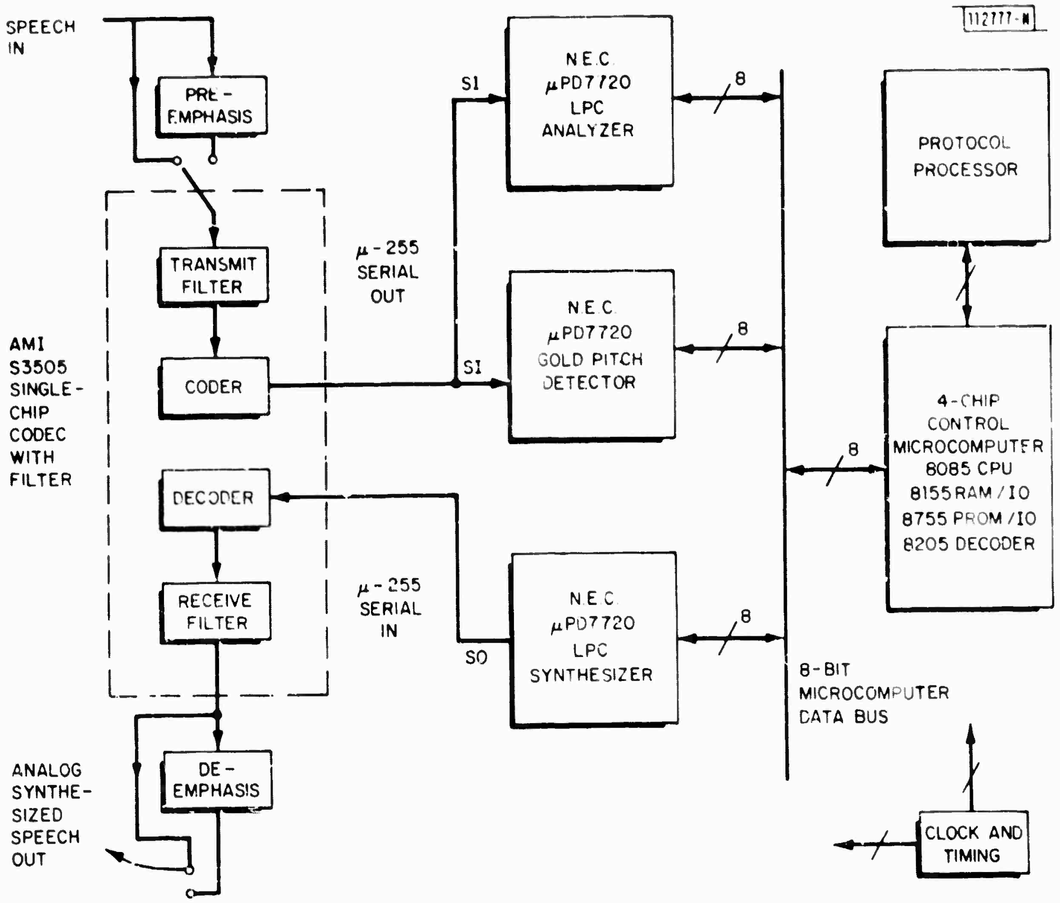

Fig. 3. N.E.C. μPD7720-based LPC vocoder.

capability, and the ability to read from and write to processor memory and registers. N.E.C. has made a copy of the EVAKIT-7720 available to Lincoln on a short-term-loan basis. An RS-232 serial communications link between the PDP-11/45 host computer and the EVAKIT has been established for '7720 program downloading as well as interactive real-time debugging. A downloader program, developed for a different application in the wideband network effort, is being used without modification as the corresponding software driver for this I/O channel. The EVAKIT interfaces to the target application simply by plugging into the DIP socket intended for the μPD7720. The Gold pitch algorithm was downloaded from the PDP-11/45 into the EVAKIT and run with the channel vocoder wirewrap board prototype. Although the pitch detector is yet to be integrated with an actual analysis-synthesis system, the resulting pitch track was displayed in real time by the LDSP giving a high level of confidence in the functionality of the microcode. Just as importantly though, this test corroborates the correctness of the Lincoln Laboratory in-house programs for N.E.C. SPI code development. The EVAKIT is expected to be an invaluable real-time debugging aid in the future, and fits well into the multi-processor test scenario described above by facilitating the debugging of a single N.E.C. device's microcode with real-time emulation of the remainder of the vocoder board by the LDSP.

C. LSI LPC VOCODER DEVELOPMENT

As previously reported, delivery of the EPROM versions of the μPD7720 has been delayed. Since there is no indication of their availability in the near future, the alternative of ordering factory-programmed ROM versions of the '7720 is being investigated. The recently proven availability of the EVAKIT for real-time verification of μPD7720 microcode in the actual application environment is a significant factor in the feasibility of such an approach. It has previously been reported that an 8-kHz full-duplex LPC vocoder could be implemented with three N.E.C. SPI devices: one for a Gold pitch processor, one for the LPC analysis, and one for the synthesizer (Fig. 3). Furthermore, it is estimated that the '7720 ROM resources are sufficient to house both the analysis and synthesizer programs and coefficients, resulting in a total of two

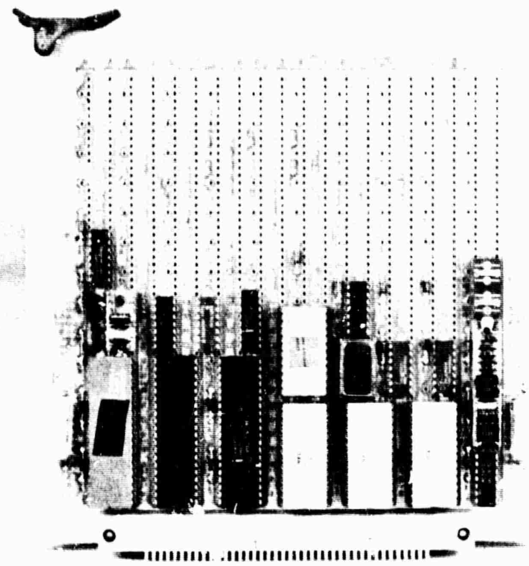

Fig. 4. N.E.C. μPD7720-based LPC vocoder.

112779-N

DEVICE	QUANTITY	POWER DISSIPATION (W)
40-PIN INTEL 8085 CPU	1	1.5
40-PIN INTEL 8185 RAM/IO	1	1.5
40-PIN INTEL 8755 PROM/IO	1	1.5
28-PIN N.E.C. μPD7720 SIGNAL-PROCESSING CHIP	3	2.7
24-PIN AMI S3505 CODEC WITH FILTER	1	0.1
14- AND 16-PIN MISCELLANEOUS DUAL-IN-LINE PACKAGES (clocks, timing, etc.)	9	1.3
DISCRETE COMPONENT CARRIERS AND SWITCHES	5	—
TOTAL	21 PACKAGES	8.6

Fig. 5. LPC package count and power dissipation.

μPD7720 masks to implement an LPC algorithm. Since a minimum number of masks would be desired in a ROM realization of a narrowband vocoder, we are actively pursuing the development of an N.E.C.-based LPC vocoder.

The detailed hardware design and prototype fabrication have been completed for a single-board N.E.C. SPI-based 8-kHz LPC full-duplex vocoder for the PVT (Fig. 4). The design requires 21 packages and approximately 16.5 sq. in. (or less than half) of the space available on the PVT Augat wire-wrap board (Fig. 5). A layout of the LPC vocoder on an Augat universal wire-wrap board indicating major functional partitioning is shown in Fig. 6. The LPC design differs from the Belgard vocoder in that it requires only three

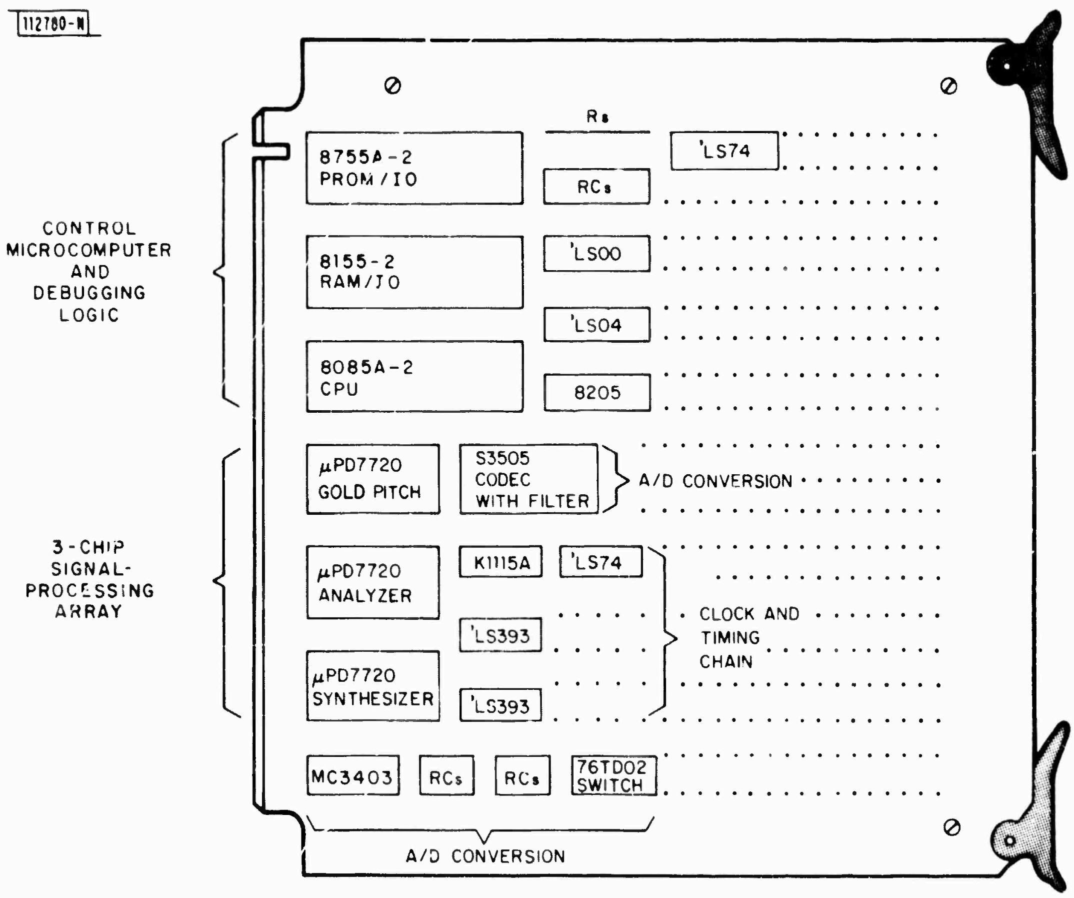

Fig. 6. Single-board (7- × 7-in.) LPC vocoder.

N.E.C. μPD7720 devices (LPC analyzer, Gold pitch detector, and synthesizer) compared with six for the channel vocoder, and that the LPC board has a simplified timing chain resulting in a reduction in the number of SSI devices. The LPC design also features a manual switch for choice of analog or digital pre- and de-emphasis to insure maximum compatibility with other LPC implementations.

The single-chip channel vocoder Gold pitch detector can be used in the LPC vocoder, but must be modified to operate at the lower sampling rate of 8 kHz. The detailed microcode for the remaining two N.E.C. processors implementing the linear predictive analyzer and synthesizer has yet to be written, although major inner loops as well as the main data structures have been benchmarked.

The real-time debugging of the LPC vocoder can proceed in a manner similar to the channel vocoder using the EVAKIT-7720 in conjunction with an LDSP emulating the remainder of the system. As a final test of the LPC implementation, it is hoped that three EVAKIT-7720's will be available for a standalone real-time emulation independent of the LDSP.

In summary, development of the N.E.C.-based channel vocoder is well advanced. The 10-kHz μPD7720 implementation of the Gold pitch detector has been simulated in real time. Delivery of EPROM versions of the '7720 in the short term is uncertain; therefore, a '7720 ROM-based realization of a narrowband vocoder is also being investigated. An LPC vocoder is expected to require only two ROM masks, and therefore is an appealing choice for initial ROM implementation. The hardware design for an 8-kHz full-duplex LPC vocoder has been completed and a single unit fabricated. Twenty-one packages are required, occupying less than half the 7- × 7-in. Augat wirewrap board. The LPC vocoder N.E.C. SPI microcode is currently being developed.

II. EMBEDDED CVSD-BASED WAVEFORM CODER

The functional and hardware design of a compact embedded data stream speech waveform coder was described in the previous Semiannual Technical Summary.* This Embedded Continuously Variable Slope Delta Modulation (ECVSD) coder has been developed to support rate-adaptive speech-flow control experiments with the Packet Voice Terminal. The units use a backbone 16-kbps CVSD coder, with additional PCM coding of the open-loop error to provide rates of 32, 48, and 64 kbps. Two ECVSD units, each on a single 7- × 7-in. PVT-compatible wirewrap card, have now been constructed and tested in conjunction with a PVT. Some necessary changes in the hardware design have been implemented, and microcode for the INTEL 8085 processor portion of the ECVSD card has been developed and tested.

The final hardware realization of the ECVSD unit includes two additional low-pass filter packages not shown in the previous report. One of these filters provides delay equalization between the CVSD path and the PCM-coded error signal. In addition, although each switched capacitor filter chip contains a pair of low-pass filters, it was found in testing that only one filter in each package is of sufficient quality for use, so that separate A/D and D/A chips are now used. Figure 7 is a board layout for the final design.

The operating microcode running in the 8085 chip set must format the CVSD and error sign ls into four separate priority frames for transfer to and from a protocol processor. The microcode is structured as an interrupt-service routine and a background routine which swaps data with a protocol processor. The interrupt-service routine is activated every 500 μs (every eight CVSD samples) and must read in a new CVSD byte as well as read out a new byte for the output speech signal. In addition, 3 bytes are read and 3 are written corresponding to the highest, middle, and least-significant bits of a 6-bit error word. The silence detector is polled when a new set of buffers is about to be loaded for output to the protocol processor. The received header

* Semiannual Technical Summary, Packet Speech Systems Technology, Lincoln Laboratory, M.I.T. (31 March 1981), DTIC AD-A104373.

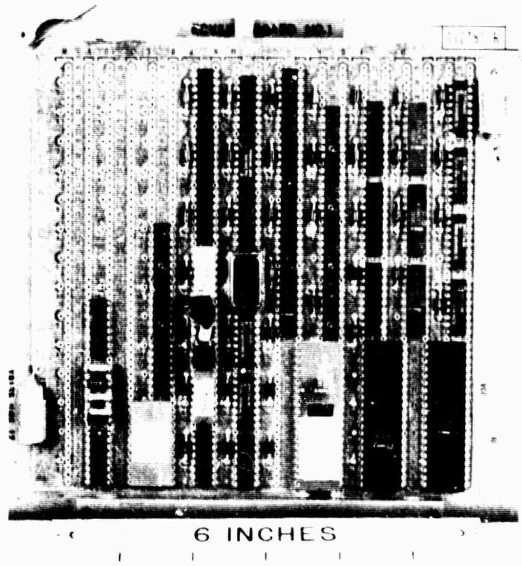

Fig. 7. Board layout for ECVSD coder.

word is tested for silence, 16-, 32-, 48-, or 64-kbps data to be used for the received speech signal. Double buffering is used for both receive and transmit data paths. The buffering structure and separation of foreground and background input/output tasks are indicated in Figs. 8(a) and (b).

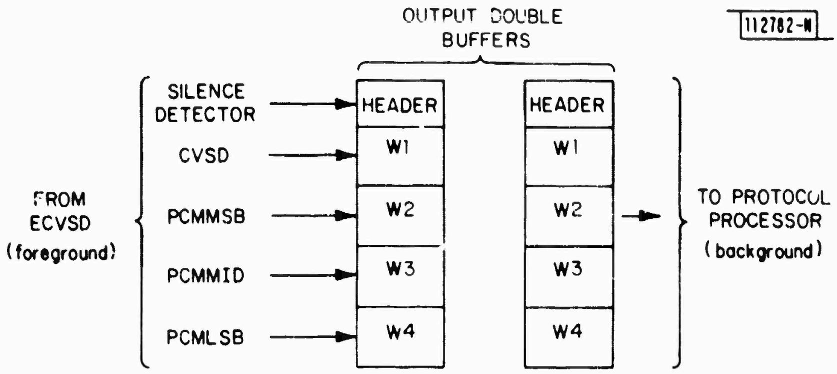

Fig. 8(a). ECVSD transmitter buffering structure.

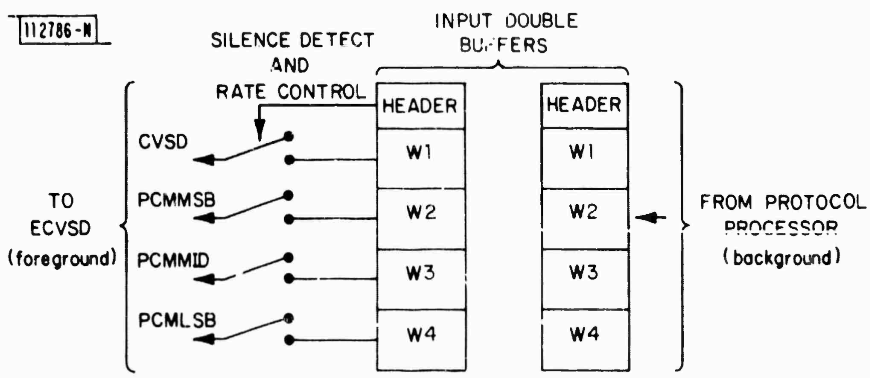

Fig. 8(b). ECVSD receiver buffering structure.

III. PACKET VOICE TERMINAL AND LOCAL ACCESS AREA

A. SUMMARY

Ten Packet Voice Terminals (PVTs), three LEXNET/Concentrator Interface units (LCIs), and six Trap Debugger units have now been assembled and debugged. A LEXNET is now in place at BBN, in addition to the LEXNETs at Lincoln and ISI. Successful speech-loop tests have been conducted at BBN using a pair of PVTs in conjunction with the Voice Funnel. An interoffice cable net has been installed at Lincoln to support LEXNET experiments. An EPROM version of the PVT memory extension card capable of supporting 32K bytes of program memory for standalone operation has been developed.

The PVT technology transfer has been initiated by setting in motion a preliminary "expression-of-interest" exercise. A telephone survey identified about 15 vendors as potential respondents to a future PVT request-for-quote. A detailed technical information package has been mailed to each. A set of approximately 115 detailed technical drawings covering both mechanical and electrical components of the PVT system has been completed, and will be made available to vendors who express serious interest in technology transfer.

The PVT software has been expanded substantially. A source-routing option allows PVT control of the packet route, including the facility for loop-backs at either the concentrator or PSAT level. An option to encapsulate stream (ST) packets in standard Internet Protocol (IP) datagram packets allows voice communication through gateways, including the Voice Funnel, which currently do not handle ST traffic. Initial integration of the PVT protocol processor with the embedded waveform coder has been achieved, and protocol software to support a variety of embedded coding data packaging and rate control options is under development. The Traffic Emulation Module (TEM) software capability of the PVT has been used for a variety of system tests; in particular, the LEXNET has been shown to be capable of supporting about 750 kbps of packet voice traffic.

B. PVT AND LEXNET HARDWARE

1. Hardware Replication and Debugging

Ten complete PVT units have been assembled and successfully debugged. Three LCI units and six Trap Debugger units have also been constructed and debugged. The decision to wire the eleventh PVT box, which was to have been a spare terminal, as an LCI unit was made to support Voice Funnel interfacing experiments at BBN. A full complement of boards for two additional PVTs has been constructed. These boards are being debugged and will serve as spares.

2. PROM Memory Extension Card

In order to turn the PVT into a standalone unit which does not require downloading of the protocol program, an EPROM version of the memory extension card has been developed. This card is a direct substitute for the RAM memory extension card, and will support up to 32K bytes of program memory using $2K \times 8$ EPROM chips. The protocol processor program was reorganized slightly so that only program, and no data or variables, resides in the memory extension; identical programs now can run in the EPROM and RAM PVTs. The PROM programmer on our HP64000 microprocessor development system is being used to "burn" programs into the EPROM card. As with the RAM version of the system, the protocol programs, written in C and A-Natural, are compiled on the PDP-11/70 using a Whitesmith compiler. We have written a program for the PDP-11/70 which converts the output of the Whitesmith compiler into a form which is usable by the HP64000 PROM programmer, and transmits the converted file to the HP64000 via TTY line. The PROM version of the PVT has been used for voice tests (see below) in conjunction with the Voice Funnel and with a local interoffice cable net at Lincoln.

3. Interoffice Cable Network (ICN)

An ICN has been established which links twenty destinations over a span of approximately 1250 ft. The network consists of a coaxial cable with coaxial tees in various office and laboratory complexes suitable for attachment to

Packet Voice Terminal cable tap boxes. These taps provide optical signal coupling to modem boards resident in the terminals. The characteristic impedance of the cable is 75 ohms, and communication between terminals is effected via baseband ETHERNET-like signaling using contention-based access strategies embodied in the Buffer Control Processor of the Packet Voice Terminals. The ICN has been used to demonstrate voice connectivity via terminals situated at remote ends of the network and at various other tapped destinations.

4. Voice Funnel Interfacing

LEXNET and PVT equipment has been delivered to BBN, and interfaced successfully to a Voice Funnel. First, a loop test was carried out between the Voice Funnel and an LCI unit. The Funnel's I/O processor successfully looped packets to the LEXNET cable. The serial rate over the connection was 750 kbps. Later, two EPROM-version PVTs were brought to BBN and duplex packet speech, transmitted over the LEXNET and looped through the Funnel, was communicated successfully between the terminals. PVT protocol software modifications required for this experiment included IP/ST encapsulation and IP source-routing, as described below.

5. LEXNET/PVT Demonstration

A number of milestones in LEXNET/PVT hardware developments were combined in a demonstration at the Wideband Meeting in May at Lincoln Laboratory. A PROM-equipped PVT was connected in a conference room at one end of the ICN cable. A second PVT, equipped with a Switched Telephone Network Interface (STNI) card developed by ISI, was located near the opposite end. The STNI card was connected to a Telephone Extension on the Lincoln PBX. In this configuration the capability was demonstrated to dial from a Lincoln Extension into the LEXNET via the STNI-equipped PVT, and to establish a PCM packet voice connection to the PVT in the conference room.

C. PVT TECHNOLOGY TRANSFER

The PVT technology transfer is being initiated by setting in motion a preliminary "expression-of-interest" exercise. A telephone survey identified

about fifteen vendors as potential respondents to a future PVT request-for-quote. A detailed technical information package has been prepared for mailing to each. Further discussions will be conducted after the prospective vendors have had an opportunity to review the material. A range of options will be explored during these discussions, including: (1) a direct copy of the current PVT, (2) a modified version of the equipment engineered to improve the packaging, and (3) a full-scale redesign of the equipment based on the functional and protocol specifications embodied in the LEXNET PVT. The main objective of this "expression-of-interest" exercise is to establish a realistic basis for estimating costs and lead times involved in a PVT technology transfer procurement, under the variety of options noted.

A set of approximately 115 detailed technical drawings covering both mechanical and electrical components of the PVT system has been completed. These drawings will be made available to prospective vendors who express serious interest in technology transfer.

D. PVT SOFTWARE

1. Dialing and Routing Options

The PVT software has been expanded to allow a caller to select among dialing sequences which cause a PVT to:

(a) Send all ST messages encapsulated in IP messages,

(b) Source-route all IP messages,

(c) Ascertain the address of its LEXNET,

(d) Choose to send all messages via either the miniconcentrator or the Voice Funnel.

The ST protocol provides for having ST messages encapsulated in IP messages so that they may pass through gateways which do not handle ST traffic. Source-routing, which is a specified option in the DoD standard Internet Protocol, allows the caller to specify what route his IP packets would follow to reach their destination. (There is currently no provision in the ST protocol for source-routing.) Gateways normally send messages to their destination via

the shortest route. By using encapsulated IP/ST messages and source-routing, traffic to another local PVT can be sent via the PSAT. Encapsulated IP/ST messages have also been used at BBN for testing the Voice Funnel since it does not handle ST in its initial implementation.

A PVT must know the address of the LEXNET it is connected to in order to provide the proper return address to a called PVT on another LEXNET. This information is provided by the miniconcentrator, which is assumed to know the identity of any LEXNETs which are attached to it. When the caller dials a call to another LEXNET, the PVT sends a short message to itself via the miniconcentrator to ascertain what LEXNET it is on. If the message does not return, the caller is given a dial tone to indicate that his call cannot be completed. The caller can also indicate whether he wishes his inter-LEXNET call to go via the Voice Funnel or via the miniconcentrator, and can select the options of source-routing and/or encapsulated IP/ST packet protocol.

2. Wideband Satellite Network (WB SATNET) Speech Loopback

About a week after the 24-25 May Wideband Meeting at Lincoln, PCM packet speech was successfully communicated (half-duplex) between two PVTs on a LEXNET, looped through the miniconcentrator the PSAT, and over the channel. The above-described PVT features of IP/ST encapsulation and source-routing were used in forcing the loopback path. BBN set up a PSAT stream to handle the packet flow, which was 50 packets/s (one way) from PVT to PVT. Since that test, experiments directed at PVT communication over the channel have been deferred while a major effort in multi-site WB SATNET system integration has been under way.

3. Speech Packetization and Buffering

A new 180-byte-per-parcel packetization protocol for PCM encoded speech has been implemented to better match the stream frame interval provided by the PSAT. To determine the stream frame interval, the PSAT takes the 3.088-Mbps clock from the satellite channel and divides it by 2^{16}. This produces a PSAT frame interval of 21.22 ms. The original 160-byte-per-parcel PCM vocoder produced a frame every 20 ms. Because the packet generation

rate was faster than the stream slot rate, about once in every twenty packets the stream would have to carry an extra packet or speech would be lost. Since a loss of 5 percent of the packets due to stream overflow was not viewed as acceptable, it was necessary to increase the stream slot size to allow the extra packet to be carried. This would not be a major problem when many speakers share the stream and therefore the extra slot, but for a single-speaker stream nearly 50 percent of the stream capacity was wasted. By putting 180 bytes in each parcel we get a parcel time of 22.5 ms, which is longer than the stream interval of 21.22 ms. This assures that there is never more than one packet waiting for each stream slot. With this combination, packets do not accumulate. Occasional slots go empty, but the wastage for a single-speaker stream is only about 5 percent.

A new smoothing algorithm allows the PVT to make more efficient use of the small amount of memory which can be accessed by the DMAs. During early test runs with the PSAT, it was noted that the buffering being used for smoothing the received speech was not sufficient to cope with the amount of delay dispersion being experienced. Code was added to the PVT to measure the time dispersion of arriving speech packets, and measurements confirmed that the capability of the PVT to cope with packet dispersion needed to be increased. To correct this difficulty, the layout of data memory in the PVT was reorganized to maximize the amount of buffering available for reconstitution of the received speech. In addition, the program was changed to allow an arbitrary number of buffers to be used (the older version had required that the number of buffers be a power of two). In the new program, a table of buffer pointers whose length must be a power of two (we are currently using 32) is maintained. The buffer to be used is determined as follows. The timestamp of the first data message in each talkspurt is used to determine the offset between it and the timestamp being used to play out parcels to the vocoder. Thereafter, the timestamp of an incoming parcel has this offset added to it, and then a reconstitution delay is added. The least-significant 5 bits of the resulting timestamp are then used to pick a buffer pointer out of the 32-pointer table. If we really had 32 buffers the table could remain fixed. However, since

we have only 16 real buffers the other 16 pointers all point to imaginary buffers in nonexistent memory. The pointer table is managed so that at all times the 16 real buffers occupy the 16 registers in the table, beginning with the buffer currently being played out to the vocoder. This automatically causes late-arriving parcels to be read into nonexistent space and therefore discarded. Data are kept on the dispersion of arriving parcels and on the number discarded.

4. Embedded Coding Protocols

An embedded CVSD-based waveform coder is now available for use with the PVTs. This embedded coder provides a speech data stream of 64 kbps in which are embedded speech data streams at 48, 32, and 16 kbps. If the data are transmitted in packets of descending priority, a gateway could discard lower-priority packets when it is unable to handle high rate transmission. The receiver will produce speech output with quality which increases with its received data rate. PVT protocol software to support a variety of embedded coding data packaging and rate control options is under development. Some alternative techniques to be investigated are described below.

The PVT could send all the data for each time frame in a single packet. The gateway could then lower the transmission rate by slicing the correct amount of data from the end of the packet. This violates the network principle that a gateway should never need to know the format of the data portion of the messages it processes. This method has been discarded.

The PVT could send four messages per time frame (i.e., 22.5 ms). Each message would contain the data for a 16-kbps slice and carry a priority designation. Retaining the highest-priority packet only is equivalent to a 16-kbps coder; the two highest-priority slices yield a 32-kbps coder, etc. This method quadruples the number of packets sent, resulting in an increase in overhead and in packet-handling load on the LEXNET and on the gateway. However, this method keeps the speech packetizing delay down at the frame interval of 22.5 ms.

The PVT could combine the four packets of data into one larger message. The ST protocol provides for the handling of such aggregated packets. Each parcel would have its own header which contained its priority. The overhead is

reduced rel e to the first option, and the PVT sends one message per time frame. Th method requires that the gateway disassemble such a message, decide what parcels to retain, and reassemble the parcels to be forwarded. This approach does not increase the number of packets on the LEXNET, but it does pose a factor-of-four increase in processing load on gateways which may well be overburdened in a heavy-load situation.

The PVT could combine data slices from adjacent time intervals such that each message contained only data of the same priority and only one header. For 64 kbps, the highest-priority data slice from four adjacent time intervals would be combined and sent as the first transmission. The next-highest-priority data slices would go out in the next transmission, etc. Each message would carry one priority designation which applied to the entire message. The gateway could easily drop messages as necessary. The packetizing delay would be increased to 90 ms, but the effect of this delay could be compensated for somewhat by beginning transmission with the first data frame which goes over threshold and combining it with the three prior "silence" frames. This would also help the transition from silence to speech. This method requires that the PVT be able to combine data from as many as four adjacent speech frames and also be able to distribute the incoming speech correctly. When a maximum rate of 48 kbps is desired, the data would be combined over three adjacent speech frames while a rate of 32 kbps would combine two frames.

Our plan is to provide a variety of options. The second or third option above will provide minimum delay to individual users, but the last option may be required to maximize gateway throughput under heavy loading. Initial rate-control tests with the ECVSD system will allow the user to specify dynamically the data rate the PVT should attempt to send. Later experiments will utilize advanced traffic-control code to be written for the gateways to implement the discarding of low-priority messages and to deliver rate-control messages to terminals.

5. Traffic Emulation and Measurements

The Traffic Emulation Module (TEM) software capability of the PVT has been used to test the system, to obtain information about system bottlenecks,

and to begin system performance measurements. The ability to source-route IP packets was added to the TEM (see discussion under PVT software) to increase its usefulness.

A set of measurements has been made to assess the load that could be handled by a LEXNET. To put the maximum load possible on the LEXNET using only two PVT's, an option was added which causes the PVTs to send their messages to a nonexistent host. An experiment with two PVTs both sending their messages to a nonexistent host has shown that the LEXNET is capable of carrying about 750 kbps of packet speech traffic, while operating at a 1.0-Mbps transmission rate on the cable.

IV. MINICONCENTRATOR

A. SUMMARY

The miniconcentrator is now operational as a flexible internet gateway, and experimental packet speech connections to LEXNETs, the WB SATNET, and the ARPANET have been demonstrated. Notable milestone experiments were: (1) transmission of PCM speech through the gateway in a 3-network configuration (two LEXNETs and WB SATNET), and (2) half-duplex PCM speech from a LEXNET through the gateway and over the WB SATNET channel (see Sec. III-D-2). With respect to the second experiment, we are working with BBN to discover and correct both a "lost packet" problem and a larger-than-expected delay dispersion which have prevented fully satisfactory speech performance through the PSAT.

Each network connection from the gateway is via a UMC-Z80 and a special network-specific interface. A previously reported problem in using the UMC-Z80 SIO chip has been resolved, but another UMC-Z80 hardware problem affecting timer interrupts to the PDP-11 has been identified.

The PDP-11 code for the gateway has been reorganized to provide a more effective structure for traffic control under heavy loading, and to allow more efficient scheduling of foreground and background tasks.

The Multiport Buffer Memory (MBM) design has been developed in further detail, and a memory test unit has been designed for the dynamic RAM chips to be used as main (MBM) memory.

B. MINICONCENTRATOR HARDWARE

Full-duplex operation of the SIO chip used in communicating between the UMC-Z80 and the LEXNET has been achieved at the intended 750-kbps clock rate. Analysis of the previously reported difficulties led to the conclusion that transmit underrun conditions were occurring due to "wait" states introduced occasionally when the Z80 attempted to access PDP-11 memory using Z80 "in" and "out" instructions. Using a test program that loops packets from the Z80 out to LEXNET and back, we have demonstrated that the transmit

underrun errors do not occur when there are no concurrent PDP-11 interactions. By replacing the Z80 "in" and "out" instructions with DMA transfers, we can avoid the "wait" states and achieve the intended transfer rate while accommodating the concurrent PDP-11 interactions. We have recoded the system to use DMA transfers for all PDP-11/UMC-Z80 data movement.

We uncovered a design error in the UMC-Z80 handling of Z80 interrupts from external IO equipment such as our 1822 DH interface board. We communicated with Associated Computer Consultants (ACC), the manufacturers of the UMC-Z80, about this error and learned that they had discussed the problem themselves and had corrected it in their more recent production cards. They have agreed to bring our old cards up to date, and we are in the process of sending them back to ACC on a one-at-a-time basis in order to keep a useful number on hand for experiments and further debugging. All but one board have been modified or are in process at this time.

We have detected another problem with the UMC-Z80. This time the difficulty is in handling interrupts to the PDP-11. The trouble occurs when the UMC-Z80 interrupt coincides with an interrupt from the internal 60-Hz clock in the PDP-11/44. When we contacted ACC, we were informed that this problem was also corrected in the newer versions of the UMC-Z80. We tried one of the newer versions and did indeed observe some improvement. Instead of failing after 3 to 5 min. of operation with our test program as we had observed with an early version, the new version lasted 15 to 20 min. We are continuing our investigation of this problem since we use a combination of the UMC-Z80 timer and PDP-11 interrupts to properly schedule the dispatching of PSAT stream packets. We have learned that people at University College, London have experienced a similar problem in their application of the UMC-Z80 and plan to keep in contact with them as we proceed with our diagnosis and correction of the problem. Meanwhile, we expect that we can circumvent this difficulty in carrying out initial speech experiments over the WB SATNET between Lincoln and ISI.

C. MINICONCENTRATOR SOFTWARE

1. Z80 Software

Previously, when the gateway program ran on our PDP-11/45 computer, communication with the ARPANET was achieved using the IMP interface hardware on the 11/45 and a software module that simulated the functions of the UMC-Z80. To get an ARPANET connection for the current 11/44 gateway, we are using a UMC-Z80 with an 1822 distant host interface and the same hardware configuration used to connect to the WB SATNET. A Z80 program and a relatively small network-specific module for the PDP-11 have been written to handle the ARPANET-specific code (e.g., management of local network headers) that the gateway must execute to interface to ARPANET. Utilizing this new configuration, speech communication via ARPANET between a LEXNET PVT at Lincoln and a speech program at ISI was successfully demonstrated.

2. PDP-11 Gateway Software

During this report period, the gateway program was extended in capabilities and flexibility.

Different versions of the gateway program are needed to encompass different combinations of networks. To satisfy this requirement, a table-driven approach was introduced for the selection of the networks which a given version of the gateway program is to handle. For example, in one version the gateway can be connected to any combination of two LEXNETs, the WB SATNET, and the ARPANET.

The handling of IP source-routing was incorporated into the gateway to provide a means for forcing the routing of traffic via selected paths. This permits the testing and measurement of traffic through paths that would otherwise not be practical.

Utilizing the above-described extensions to the gateway program, an important milestone was achieved in May 1981 by the transmission of PCM-encoded speech in a 3-network configuration at Lincoln. The path for speech in this experiment was as follows. PCM-encoded speech originated at a PVT

in LEXNET 1. From there, it went via the UMC-Z80 to the gateway program running on the PDP-11/44. Using IP source-routing (supplied by the PVT), the gateway forwarded the speech to the PSAT which returned it back to the PDP-11/44. This time the gateway transmitted the speech (as specified by the IP source-routing) via another UMC-Z80 to a PVT on LEXNET 2. The configuration for this experiment is shown in Fig. 9. As indicated, two PVTs were connected to each LEXNET; a pair of simultaneous full-duplex PCM conversations looped through the gateway was demonstrated at the May 1981 Wideband Meeting.

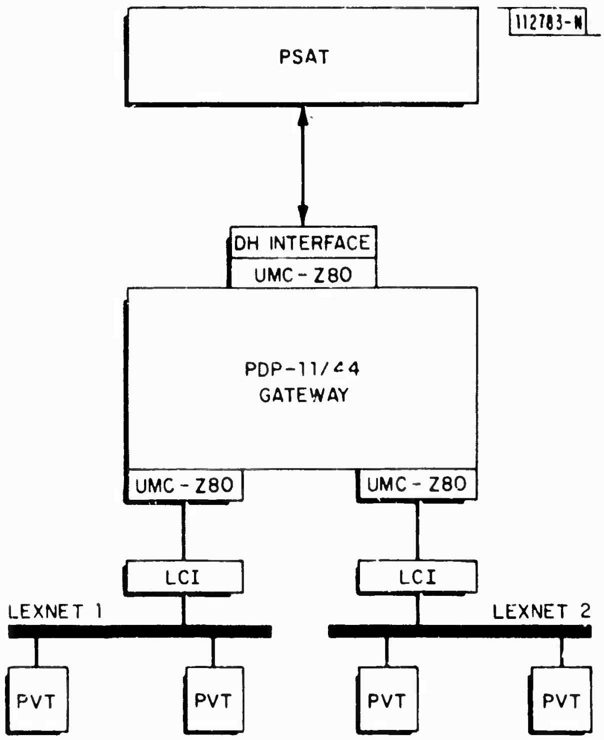

Fig. 9. Gateway configuration for 3-network packet speech experiment.

The PDP-11 code for the gateway program has been reorganized to provide a more effective structure for traffic control under heavy loading, and to allow more efficient scheduling of foreground (e.g., packet forwarding) and background (e.g., ST connection management) tasks. The gateway also now provides a finer grain time base to trigger the forwarding of speech stream

packets. The program now runs in three processes that share a common address space, with each process running at a different priority.

Running at the highest priority is a clock process that provides timing and supervisory functions. This process receives timing signals from the PDP-11's internal 16.7-ms (60-Hz) clock and from hardware clocks on the UMC-Z80's. The packet forwarding process (second level of priority) is currently activated every 11 ms on a periodic trigger derived from a UMC-Z80 clock. Supervisory and control functions are triggered by the PDP-11 clock. For example, the PDP-11 clock is currently used to verify that the Z80 which is providing the basic time base is functioning correctly. If the Z80 time base fails, the clock process can call on another of the Z80's attached to the gateway or use a 60-Hz clock signal itself. The PDP-11 clock also provides a triggering mechanism for supervisory processes which can examine packet queues, and for traffic-control processes which can take appropriate action (e.g., discard low-priority packets, or shut down selected ST connections) in case of observed overload. We expect to be experimenting with control mechanisms of this type in conjunction with the use of streams in the PSAT.

Previous versions of the gateway program had used the PDP-11 60-Hz clock to trigger all functions, including packet forwarding. In addition to providing a more effective control structure, the current gateway provides a faster forwarding trigger clock (not available in the PDP-11 hardware) via the UMC-Z80. This will allow the forwarding process to be more effectively matched to the 21.2-ms PSAT frames.

The forwarding process performs the sorting, table lookup, and dispatching functions associated with packet forwarding. If this process finds an ST control message, a local net (not internet) packet, or any other message directed to the gateway itself, it dispatches it to the lowest-priority process for handling.

The lowest-priority process acts as a background program handling all tasks that do not have critical timing requirements, e.g., ST connection management functions and some keyboard/typewriter interactions. Separation of these functions into a lower-priority process insures that real-time tasks will not be unnecessarily delayed by background tasks. Since the forwarding and

background processes are not synchronized, we use the same circular buffering convention for communicating between them that we use for communicating between the PDP-11 and the UMC-Z80.

As indicated in the above description, our work to date in the traffic-control area has been concerned with the design and implementation of a number of generally low-level mechanisms for sensing network and processor loads, for protecting resources by discarding traffic, and for negotiating for communication capacity between packet voice hosts and gateways, and between gateways and the PSATs that control SATNET stream resources. Further design work is needed on low-level mechanisms to cope with overloads of data traffic as well as higher-level algorithms to balance loads and resources and to cope with link and node outages by adjusting routing tables. Since the wideband network has not yet reached a status that can support traffic-control experiments, we will defer a more detailed presentation of these mechanisms until we have some experimental evidence of their effectiveness in controlling network traffic.

Numerous efforts have been carried out in the area of support software for the miniconcentrator.

The downloader program runs on a host computer providing communication with the miniconcentrator computer via an RS-232 TTY line. During this report period, the downloader was extended in several important ways. Heretofore, the program ran only as a user program under the UNIX operating system. Since ISI regularly runs the EPOS operating system on its host computer, the downloader was modularized to permit a version to be generated which runs as a user program in the EPOS environment. A second extension provides for a canned script file to be used as a means for automated execution of commands to the downloader or to the object machine. This permits the 3-network gateway program to be invoked in a relatively simple automated manner. Other uses for this automation could include examination and/or modification of groups of parameters in an error-free manner. In addition, a new command was introduced for retrieving files from the downline computer; this complements the download command and allows for convenient backup and verification of file integrity.

Finally, the versatility and power of the downloader have been made apparent by the development of a new use for it, viz., to communicate (see Sec. I) between the PDP-11/45 and the EVAKIT-7720 real-time simulator/debugger for the NEC µPD7720 signal-processing interface chip. No software changes to the downloader were needed for this application.

The EPOS system has been brought up on the SRI PDP-11/44 computer by communicating with this PDP-11 computer from a "host" SRI computer running the UNIX operating system. These activities were carried out remotely from Lincoln via the ARPANET. In a joint venture between Lincoln and ISI personnel, we have evolved a library of routines which enables one to write programs in the C language in UNIX for execution under EPOS. The development of this library has progressed to the point where the compiling and linking of a program destined for EPOS are nearly as convenient as one destined for UNIX.

D. MULTIPORT MEMORY DESIGN

The architectural configuration for the MBM is shown in Fig. 10. The Z80 ports in the miniconcentrator will be attached to LEXNET, PSAT, and ARPANET complexes. Various aspects of the detailed design to implement the Bus Arbitration and Memory Management logic are under study. At issue is the possibility of implementing at least some portion of these functions in a microprocessor.

The design of the MBM was modified to allow the use of full 16-word data strings rather than 15-word strings and an address pointer to the next string. It was decided to implement the string pointer memory as an adjunct to the main memory complex rather than to incorporate it into the main memory. Thus, the string pointers will be resident in an auxiliary memory specifically dedicated to a location list of consecutive 16-word blocks with block truncators. This memory will share a common address and data bus with the main memory complex. Control of the MBM will be effected through the use of a finite-state machine. The candidate structure to house this machine and the two memory complexes is shown in Fig. 11. The auxiliary string pointer

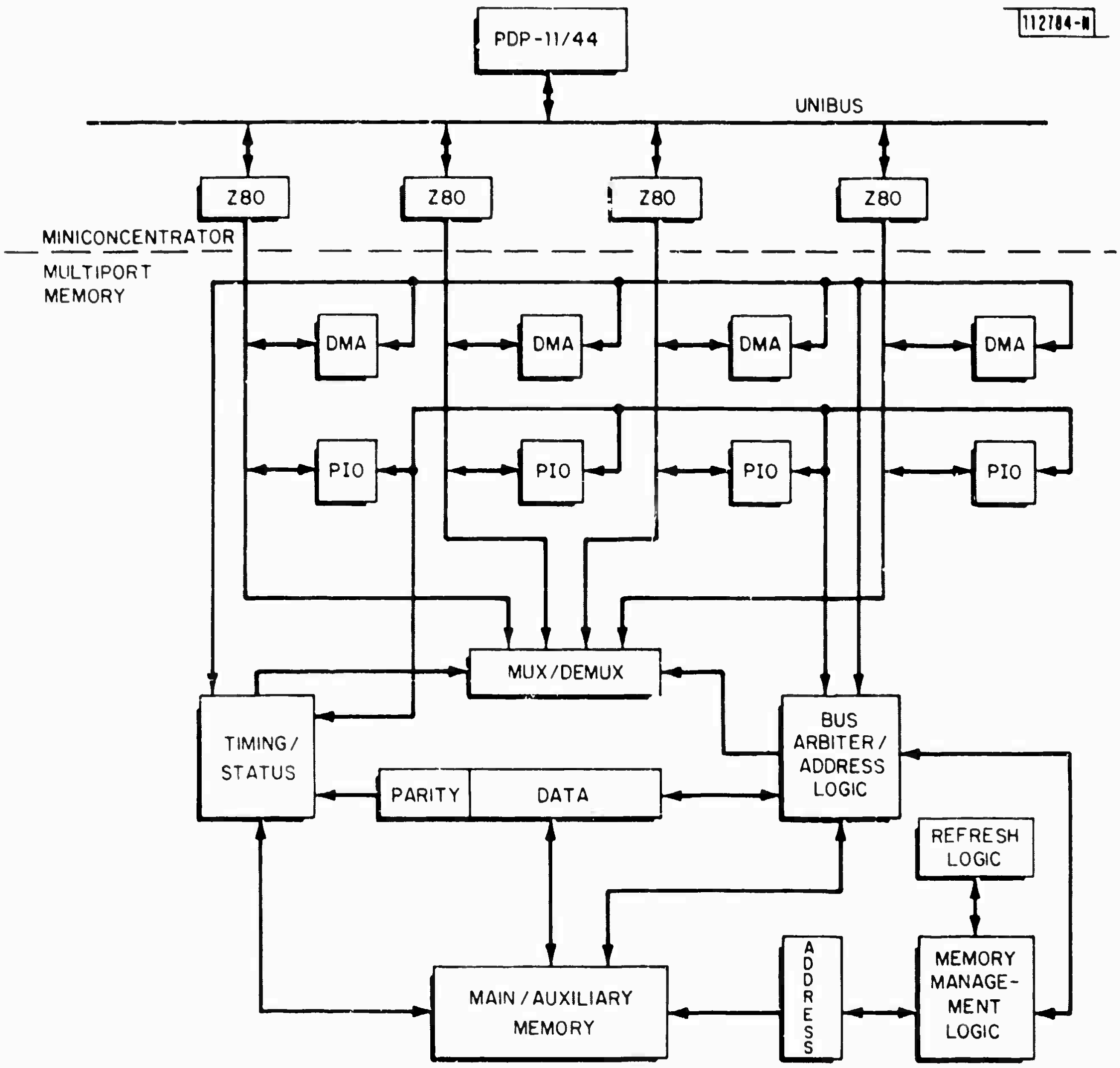

Fig. 10. Architectural configuration for MBM.

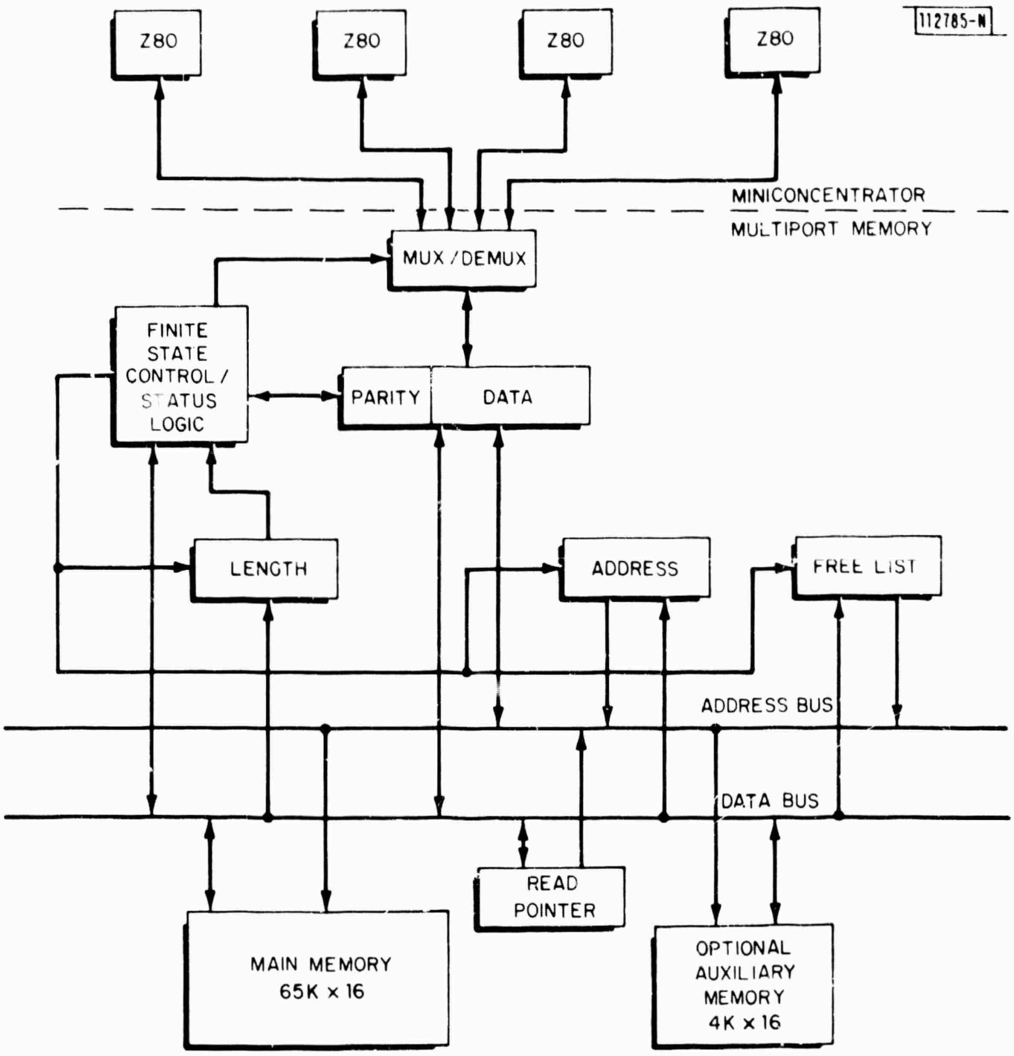

Fig. 11. Candidate structure for finite-state machine.

memory of size 4K × 16 will be composed of static RAM chips. The advantages of this design change are threefold:

(1) The address manipulations will be considerably simplified by the use of 16-word data blocks.

(2) The reliability requirements for the string pointer main memory are considerably more stringent than for the main memory. Thus, the use of static RAMs for the auxiliary memory will reduce the chance of soft memory errors inherent in the denser dynamic RAMs.

(3) The full 64K of dynamic RAM will be available for the main data memory as a shared data resource.

A memory test unit is under development for testing dynamic RAM parts for the main memory complex. An array of 64K × 8 will be assembled in the tester, and it will be used in conjunction with the LDSP to apply various test patterns and to record results. The tester will be used to ascertain which parity and/or error correction and detection configuration will be employed in the main memory unit. It is expected that soft memory errors will be encountered with the 64K dynamic RAM chips due to alpha particle effects, but no quantitative data are available to determine to what degree such errors will adversely affect the total systems environment of the MBM.

V. EXPERIMENT PLANNING AND COORDINATION

Lincoln has performed an active role in the coordination of activities aimed at checking out and fully stabilizing the wideband satellite link. Through the cooperation of personnel at the WB SATNET nodes as well as Western Union and COMSAT, serious problems of satellite channel quality and reliability have been corrected. An intermittent RF interference problem at ISI is still an issue, as described below. The current status may be summarized as follows. Earth-station equipment has been installed and is operating at all four sites. Two 48-h tests of the channel (with Lincoln and ISI transmitting, respectively) were carried out in July to test the reliability of operation at bit error rates satisfying the specified maximum of 5×10^{-3}. The Scientific-Atlanta continuous-mode test modems were used at the two sites. In general, the results of the tests were satisfactory, although a number of dropouts occurred during ISI's transmission period; these were later traced to faulty operation of the frequency upconverter which was subsequently repaired.

Prior to the 48-h tests, Lincoln had informally served as a coordination center for usage of satellite channel time by Western Union and the various participants in the program. With the improving stabilization of the system, it became reasonable to transfer channel usage management to BBN, the organization which will routinely exercise this control after the system reaches operational status. This transfer was done in late July.

Reliability of the transmitter amplifiers has been a problem, although it appears that recent corrective action by their manufacturer (i.e., elimination of a thermally sensitive component in the high-voltage power supply) will improve the reliability to a satisfactory level. PSATs and ESIs have been installed at DCEC, ISI, and Lincoln, and each site has demonstrated satisfactory satellite loop testing from its PSAT and back to itself. A sequence of subsystem bugs and interface problems has prevented successful completion of multi-site PSAT testing and, therefore, the system cannot yet sustain regular packet communication between host computers at multiple sites. Correction of certain of these problems in the closing days of FY 81 made it possible to conduct two-site tests (between Lincoln and ISI) with reasonable reliability, and heavy

emphasis has been placed on such operation. A serious remaining obstacle is RF interference of unknown origin at ISI which causes random bursts of errors severe enough to result in frequent system crashes. Efforts are in progress to identify and remove the source of this interference.

A major Wideband Experiment coordination and information meeting was organized by Lincoln on 14-15 May 1981, at Lexington. A summary of the meeting, designated W-Note-29, has been distributed to all the participating organizations.

It was observed at the Wideband meeting that the project had evidently matured and advanced to the point at which it no longer makes good sense to convene general meetings of the entire community of participants. Instead, it would be appropriate from now on to meet in small special-interest working groups when necessary to address specific problems. Such a meeting was called by DARPA for 28 September 1981; it involved BBN, Western Union, COMSAT, Lincoln, ISI, and DCEC. Its purpose was to discuss problems and issues in bringing the satellite channel to full operational status, as well as operating it in its normal mode after such status is achieved. At the meeting it was concluded: that Lincoln would assume a more active system engineering role in resolving the RFI problem at ISI, in making the channel operational, and in seeking to improve its performance; that BBN would exercise management of the WB SATNET in a manner similar to the ARPANET; and that BBN and Lincoln would jointly produce a report outlining practical criteria for deciding whether the WB SATNET is operating satisfactorily and defining the elements of operational control and management of the network.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE — READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ESD-TR-81-302
2. GOVT ACCESSION NO.: AD A112045
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): Packet Speech Systems Technology
5. TYPE OF REPORT & PERIOD COVERED: Semiannual Technical Summary, 1 April - 30 September 1981
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Clifford J. Weinstein and Peter E. Blankenship
8. CONTRACT OR GRANT NUMBER(s): F19628-80-C-0002
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Lincoln Laboratory, M.I.T., P.O. Box 73, Lexington, MA 02173
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: ARPA Order 3673, Program Element Nos. 62708E/61101E, Project Nos. 1D10/1T10
11. CONTROLLING OFFICE NAME AND ADDRESS: Defense Advanced Research Projects Agency, 1400 Wilson Boulevard, Arlington, VA 22209
12. REPORT DATE: 30 September 1981
13. NUMBER OF PAGES: 46
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Electronic Systems Division, Hanscom AFB, Bedford, MA 01731
15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report): Approved for public release; distribution unlimited.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):
18. SUPPLEMENTARY NOTES: None
19. KEY WORDS (Continue on reverse side if necessary and identify by block number): packet speech; network speech; voice conferencing; ARPANET; SATNET; LEXNET; digital channel vocoder
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes work performed on the Packet Speech Systems Technology Program sponsored by the Information Processing Techniques Office of the Defense Advanced Research Projects Agency during the period 1 April through 30 September 1981.